
Inhaltsverzeichnis

| | | |
|----------|--|----|
| 1 | Einführung | 1 |
| 1.1 | Begriffsklärungen | 1 |
| 1.2 | Motivation und Anwendungen | 4 |
| 1.3 | Vom (Un-)Sinn des Testens | 7 |
| 1.4 | Beispiele nichtsequentieller Algorithmen | 9 |
| 1.5 | Nebenläufigkeit und der informelle Prozessbegriff | 12 |
| 1.6 | Konflikte beim Zugriff auf gemeinsame Daten | 15 |
| 1.7 | Atomare Anweisungen | 19 |
| 1.8 | Kritische Abschnitte und Sperrsynchrisation | 21 |
| 1.9 | Prozesszustände | 22 |
| 1.9.1 | Die Zustandsübergänge in C, Java und Go | 24 |
| 1.9.2 | Beispiel in C, Java und Go | 31 |
| | Literatur | 33 |
| | | |
| 2 | Schlösser | 35 |
| 2.1 | Spezifikation von Schlössern | 35 |
| 2.2 | Schlösser in C, Java und Go | 37 |
| 2.2.1 | Schlösser in C | 37 |
| 2.2.2 | Schlösser in Java | 38 |
| 2.2.3 | Schlösser in Go | 39 |
| 2.3 | Schlösser auf der Basis unteilbarer Maschineninstruktionen | 40 |
| 2.3.1 | Test and Set | 40 |
| 2.3.2 | Compare and Swap | 43 |
| 2.3.3 | Exchange | 44 |
| 2.3.4 | Decrement | 45 |
| 2.3.5 | Fetch and Add | 46 |
| 2.3.6 | Das Zählerproblem | 47 |
| 2.3.7 | Wertung des Einsatzes von Maschineninstruktionen | 47 |
| 2.4 | Schlossalgorithmen für 2 Prozesse auf Hochsprachenebene | 49 |
| 2.4.1 | Algorithmus von PETERSON | 52 |
| 2.4.2 | Algorithmus von KESSELS | 55 |

| | | |
|----------|---|-----------|
| 2.4.3 | Algorithmus von DEKKER | 56 |
| 2.4.4 | Algorithmus von DORAN und THOMAS | 57 |
| 2.4.5 | Algorithmus von HYMAN | 58 |
| 2.5 | Schlossalgorithmen für mehrere Prozesse | 59 |
| 2.5.1 | Tiebreaker-Algorithmus von PETERSON | 59 |
| 2.5.2 | Algorithmus von DIJKSTRA | 61 |
| 2.5.3 | Algorithmus von KNUTH | 63 |
| 2.5.4 | Algorithmus von HABERMANN | 65 |
| 2.5.5 | Ticket-Algorithmus | 66 |
| 2.5.6 | Bäckerei-Algorithmus von LAMPORT | 67 |
| 2.5.7 | Algorithmus von KESSELS für n Prozesse | 69 |
| 2.5.8 | Algorithmus von MORRIS | 71 |
| 2.5.9 | Algorithmus von SZYMANSKI | 74 |
| | Literatur | 78 |
| 3 | Pakete, Interfaces und abstrakte Datentypen | 81 |
| 3.1 | Die Rolle von Paketen | 81 |
| 3.1.1 | Pakete nur als Interfaces | 83 |
| 3.2 | Alle Quelltexte aus diesem Buch im nUniversum-Paket | 84 |
| 3.3 | Das Paket Object | 84 |
| 3.3.1 | Any | 84 |
| 3.3.2 | Interfaces zur Beschreibung von Objekten | 85 |
| 3.3.3 | Das Interface des Pakets | 88 |
| 3.3.4 | Warteschlangen als abstrakter Datentyp | 89 |
| 3.3.5 | Eine Warteschlange als abstraktes Datenobjekt | 92 |
| 3.3.6 | Beschränkte Puffer | 93 |
| 3.4 | Zur Problematik von Verweisen | 94 |
| | Literatur | 95 |
| 4 | Semaphore | 97 |
| 4.1 | Nachteile der Implementierung von Schlössern | 97 |
| 4.2 | DIJKSTRAS Ansatz | 98 |
| 4.3 | Binäre Semaphore | 99 |
| 4.3.1 | Äquivalenz von Schlössern und binären Semaphore | 100 |
| 4.3.2 | Algorithmus von UDDING | 101 |
| 4.4 | Puffer im nichtsequentiellen Fall | 103 |
| 4.5 | Allgemeine Semaphore | 106 |
| 4.5.1 | Spezifikation allgemeiner Semaphore | 106 |
| 4.5.2 | Entwicklung einer korrekten Implementierung | 108 |
| 4.6 | Unbeschränkte Puffer und der schlafende Barbier | 112 |
| 4.7 | Konstruktion allgemeiner Semaphore aus binären | 115 |
| 4.7.1 | Repräsentation | 116 |

| | | |
|----------|--|------------|
| 4.7.2 | Naiver (falscher) Ansatz | 116 |
| 4.7.3 | Korrektur und Konsequenz | 117 |
| 4.7.4 | Algorithmus von BARZ | 119 |
| 4.8 | Semaphore in C, Java und Go | 120 |
| 4.8.1 | Semaphore in C | 120 |
| 4.8.2 | Semaphore in Java | 121 |
| 4.8.3 | Semaphore in Go | 123 |
| 4.9 | Additive Semaphore | 123 |
| 4.9.1 | Mehrfache Semaphore | 125 |
| 4.10 | Barrierensynchronisation | 125 |
| 4.11 | Kürzeste Anforderungen zuerst | 127 |
| 4.12 | Das Leser-Schreiber-Problem | 128 |
| 4.12.1 | Das 1. Leser-Schreiber-Problem | 129 |
| 4.12.2 | Das 2. Leser-Schreiber-Problem | 130 |
| 4.12.3 | Prioritätenanpassung | 131 |
| 4.12.4 | Implementierung mit additiven Semaphoren | 134 |
| 4.12.5 | Effiziente Implementierung in Go | 134 |
| 4.13 | Das Links-Rechts-Problem | 134 |
| 4.14 | Die speisenden Philosophen | 135 |
| 4.15 | Das Problem der Zigarettenraucher | 140 |
| 4.16 | Implementierung von Semaphoren | 144 |
| 4.16.1 | Das Konvoi-Phänomen | 145 |
| | Literatur | 146 |
| 5 | Der Staffelstab-Algorithmus | 147 |
| 5.1 | Entwicklung des Problems | 147 |
| 5.1.1 | Der Staffelstab von ANDREWS | 148 |
| 5.2 | Das Leser-Schreiber-Problem | 150 |
| 5.3 | Das 2. Links-Rechts-Problem | 154 |
| | Literatur | 156 |
| 6 | Universelle kritische Abschnitte | 157 |
| 6.1 | Grundidee und Konstruktion | 157 |
| 6.1.1 | Spezifikation | 159 |
| 6.1.2 | Implementierung | 160 |
| 6.2 | Semaphore | 163 |
| 6.3 | Der schlafende Barbier | 164 |
| 6.4 | Das Leser-Schreiber-Problem | 164 |
| 6.5 | Das Links-Rechts-Problem | 167 |
| 6.6 | Universelle kritische Ressourcen | 169 |
| 6.7 | Die speisenden Philosophen | 172 |
| 6.8 | Das Problem der Zigarettenraucher | 174 |

| | | |
|----------|--|-----|
| 7 | Fairness | 177 |
| 7.1 | Schwache vs. starke Fairness | 177 |
| 8 | Verklemmungen | 181 |
| 8.1 | Charakterisierung | 181 |
| 8.1.1 | Einfache Beispiele | 182 |
| 8.2 | Gegenmaßnahmen | 184 |
| 8.2.1 | Ausschluss | 184 |
| 8.2.2 | Erkennung und Auflösung | 185 |
| 8.2.3 | Vermeidung | 186 |
| 8.2.4 | Der Bankiers-Algorithmus | 187 |
| 8.3 | Verklemmungswahrscheinlichkeit | 190 |
| 8.4 | Wertung der Gegenmaßnahmen | 192 |
| | Literatur | 194 |
| 9 | Monitore | 195 |
| 9.1 | Charakterisierung von Monitoren | 195 |
| 9.1.1 | HOARES Ansatz | 196 |
| 9.1.2 | Virtuelle Monitore in Go | 197 |
| 9.2 | Bedingungsvariable | 198 |
| 9.3 | Monitore in C, Java und Go | 203 |
| 9.3.1 | Monitore in C | 203 |
| 9.3.2 | Monitore in Java | 205 |
| 9.3.3 | Monitore in Go am Beispiel Konto | 207 |
| 9.4 | Der beschränkte Puffer | 209 |
| 9.5 | Das Leser-Schreiber-Problem | 210 |
| 9.6 | Signal-Semantiken | 212 |
| 9.6.1 | Signalisieren und fortfahren | 212 |
| 9.6.2 | Signalisieren und warten | 213 |
| 9.6.3 | Signalisieren und vorrangig warten | 213 |
| 9.6.4 | Präemptive vs. nicht präemptive Semantik | 213 |
| 9.6.5 | Vergleichende Wertung der Signal-Semantiken | 214 |
| 9.6.6 | Ein Semaphor als Monitor | 215 |
| 9.6.7 | Barrierensynchronisation | 217 |
| 9.7 | Rundruf in C, Java und Go | 218 |
| 9.7.1 | Rundruf in C | 219 |
| 9.7.2 | Rundruf in Java | 219 |
| 9.7.3 | Rundruf in Go | 220 |
| 9.8 | Der schlafende Barbier: Das Barbieren als Rendezvous | 220 |
| 9.9 | Prioritätsregelungen | 222 |
| 9.9.1 | Der HOARESche Wecker | 222 |
| 9.9.2 | Kürzeste Anforderungen zuerst | 224 |

| | | |
|-----------|---|------------|
| 9.10 | Äquivalenz von Semaphore- und Monitorkonzept | 224 |
| 9.11 | Zur Implementierung des Monitorkonzepts | 228 |
| 9.12 | Zur Problematik geschachtelter Monitorkonzepte | 229 |
| | Literatur | 230 |
| 10 | Universelle Monitore | 233 |
| 10.1 | Die Grundidee | 233 |
| 10.1.1 | Spezifikation | 234 |
| 10.1.2 | Implementierung | 235 |
| 10.2 | Konditionierte universelle Monitore | 237 |
| 10.2.1 | Spezifikation | 238 |
| 10.2.2 | Implementierung | 238 |
| 10.3 | Semaphore | 239 |
| 10.4 | Konto | 241 |
| 10.5 | Beschränkte Puffer | 242 |
| 10.6 | Der schlafende Barbier | 243 |
| 10.7 | Barrierensynchronisation | 244 |
| 10.8 | Das Leser-Schreiber-Problem | 245 |
| 10.9 | Das Links-Rechts-Problem | 246 |
| 10.10 | Die speisenden Philosophen | 247 |
| 10.11 | Die Zigarettenraucher | 249 |
| 11 | Botschaftenaustausch | 253 |
| 11.1 | Kanäle und Botschaften | 253 |
| 11.1.1 | Syntax des Botschaftenaustauschs in Go | 255 |
| 11.1.2 | Synchroner Botschaftenaustausch mit asynchronem | 258 |
| 11.2 | Asynchrone Kommunikation | 258 |
| 11.2.1 | Semaphore | 258 |
| 11.2.2 | Beschränkte Puffer | 260 |
| 11.2.3 | Die Zigarettenraucher | 260 |
| 11.3 | Netzwerke von Filtern | 261 |
| 11.3.1 | CAESARS geheime Botschaften | 261 |
| 11.3.2 | Das Sieb des ERATOSTHENES | 263 |
| 11.3.3 | Mergesort | 264 |
| 11.3.4 | Die speisenden Philosophen | 266 |
| 11.4 | Selektives Warten | 268 |
| 11.5 | Das Kunden-Anbieter-Paradigma | 270 |
| 11.6 | Synchrone Kommunikation | 271 |
| 11.6.1 | Semaphore | 271 |
| 11.6.2 | Beschränkte Puffer | 272 |
| 11.6.3 | Das Leser-Schreiber-Problem | 274 |
| 11.6.4 | Das Links-Rechts-Problem | 275 |

| | | |
|-----------|--|------------|
| 11.7 | Bewachtes selektives Warten | 276 |
| 11.7.1 | Semaphore | 278 |
| 11.7.2 | Beschränkte Puffer | 278 |
| 11.7.3 | Das Leser-Schreiber-Problem | 279 |
| 11.7.4 | Das Links-Rechts-Problem | 280 |
| 11.8 | Äquivalenz von Botschaftenaustausch und Semaphorkonzept | 281 |
| | Literatur | 282 |
| 12 | Vergleich der bisherigen Sprachkonstrukte | 285 |
| 12.1 | Schlösser | 285 |
| 12.2 | Semaphore..... | 285 |
| 12.3 | Monitore..... | 286 |
| 12.4 | Botschaftenaustausch | 286 |
| 13 | Netzweiter Botschaftenaustausch | 287 |
| 13.1 | Kanäle im Netzwerk | 287 |
| 13.1.1 | Technische Aspekte (in C) | 288 |
| 13.2 | Realisierung in Go | 289 |
| 13.3 | 1:1-Netzkanäle zwischen Prozessen auf beliebigen Rechnern..... | 289 |
| 13.3.1 | Einfaches Beispiel..... | 294 |
| 13.4 | Verteilte Schlösser nach RICART/AGRAWALA | 294 |
| | Literatur | 302 |
| 14 | Universelle ferne Monitore | 303 |
| 14.1 | Erweiterung der Netzkanäle auf den Fall 1:n | 303 |
| 14.2 | Konstruktion der fernen Monitore | 306 |
| 14.2.1 | Spezifikation | 306 |
| 14.2.2 | Implementierung | 308 |
| 14.3 | Korrektheit | 310 |
| 14.4 | Verteilte Semaphore | 310 |
| 14.5 | Verteilte Warteschlangen und beschränkte Puffer | 311 |
| 14.6 | Verteiltes Leser-Schreiber- und Links-Rechts-Problem..... | 312 |
| 14.7 | Konto | 314 |
| | Literatur | 316 |
| 15 | Netzwerke als Graphen | 317 |
| 15.1 | Graphen | 317 |
| 15.1.1 | Definition des Graphenbegriffs | 318 |
| 15.2 | Realisierung in Go | 319 |
| 15.2.1 | Spezifikation..... | 320 |
| 15.2.2 | Implementierung | 323 |

| | | |
|-----------|--|------------|
| 15.3 | Adjazenzmatrizen | 327 |
| 15.3.1 | Spezifikation | 328 |
| 15.3.2 | Implementierung | 329 |
| 15.4 | Verteilte Graphen | 333 |
| 15.4.1 | Spezifikation | 333 |
| 15.4.2 | Implementierung | 333 |
| 15.5 | Beispiele | 336 |
| 15.5.1 | Bildschirmausgabe | 338 |
| Literatur | | 340 |
| 16 | Pulsschlag-Algorithmen | 341 |
| 16.1 | Die Grundidee | 341 |
| 16.2 | Kennenlernen des Netzwerks | 342 |
| 16.3 | Voraussetzungen für die Realisierung in Go | 344 |
| 16.4 | Matrizenbasierte Lösung | 345 |
| 16.5 | Graphenbasierte Lösungen | 347 |
| 16.5.1 | Mit Kenntnis des Durchmessers des Netzwerkgraphen | 347 |
| 16.5.2 | Ohne globale Kenntnisse | 348 |
| Literatur | | 350 |
| 17 | Traversierungsalgorithmen | 351 |
| 17.1 | Voraussetzungen für die Realisierung in Go | 351 |
| 17.2 | Verteilte Tiefensuche | 354 |
| 17.2.1 | Weitergabe des Spannbaums an alle Prozesse | 360 |
| 17.2.2 | Realisierung mit fernen Monitoren und Spannbaumweitergabe | 362 |
| 17.3 | Algorithmus von AWERBUCH | 365 |
| 17.3.1 | Realisierung mit fernen Monitoren | 366 |
| 17.3.2 | Weitergabe des Spannbaums an alle Prozesse | 368 |
| 17.3.3 | Algorithmus von HÉLARY/RAYNAL | 370 |
| 17.4 | Konstruktion eines Rings | 373 |
| 17.4.1 | Weitergabe des Rings an alle Prozesse | 374 |
| 17.5 | Verteilte Breitensuche | 375 |
| 17.5.1 | Realisierung mit fernen Monitoren | 382 |
| 17.5.2 | Realisierung mit fernen Monitoren und Spannbaumweitergabe | 385 |
| Literatur | | 388 |
| 18 | Auswahlalgorithmen | 389 |
| 18.1 | Grundlegendes | 389 |
| 18.2 | Voraussetzungen für die Realisierung in Go | 390 |
| 18.3 | Algorithmus von CHANG/ROBERTS | 391 |

| | | |
|------|---|-----|
| 18.4 | Algorithmus von HIRSCHBERG/SINCLAIR..... | 393 |
| 18.5 | Algorithmus von PETERSON | 399 |
| 18.6 | Auswahl mit Tiefensuche..... | 403 |
| | Literatur | 403 |
| | Anhang A: Weiterführende Literatur | 405 |
| | Sachverzeichnis | 409 |